

# Pack Graphite

## Description du modèle

Une fois sur l'hôte, le modèle va créer un check **shinken-broker-module-event-manager-writer** pour chaque webui présent sur le démon Broker sur l'élément supervisé.

### Résultat

Le résultat du check donne le statut de l'écriture des données de l'event manager.

Le résumé sur la dernière minute contient :

- Le nombre d'événements écrits dans la dernière minute
- Le nombre de broks écrit dans la dernière minute avec le temps de traitement moyen d'un brok

#### Résultat

? Unknown Attachment

#### Brok

Les broks sont les informations de chaque vérification envoyée par le Scheduler au Broker

### Résultat long

Le résultat long donne le détail des informations traitées par le module.

La partie **Global** contient :

- Le nombre d'éléments gérés par l'event manager
- Un résumé sur la dernière minute de l'activité du module (voir ci-dessus : résultat)

Les parties **Worker** contiennent par worker :

- Le nombre d'éléments gérés par workers
- Un résumé de l'activité sur worker
- La charge du worker : C'est à dire le temps que le worker a effectivement travaillé sur la dernière minute
  - Exemple : si sur la dernière minute le module a reçu 5000 broks et qu'il a mis 10ms par broks cela fera  $(5000 * 0.01) / 60 = 0.83$  soit 83% de charge.

La partie **Database** contient :

- Le nombre de jours durant lequel sont gardés les événements. Au delà de cette limite les événements sont supprimés.
- Le nombre d'événements sauvegardés et la taille de la base.
- Date du dernier événement sauvegardé.

#### Résultat long

? Unknown Attachment

### Métriques

Voici les métriques disponibles :

Nom de la métrique	Description
worker_[X]_load_in_last_min	Charge du worker, entre 0 et 1.
worker_[X]_event_write_in_last_min	Nombre d'événements écrits sur la dernière minute par le worker.
worker_[X]_brok_handle_in_last_min	Nombre de broks traités sur la dernière minute par le worker.
global_event_write_in_last_min	Nombre total d'événements écrits sur la dernière minute
global_brok_handle_in_last_min	Nombre total de broks traités sur la dernière minute
total_base_size	Taille de la base en octet
total_event_number	Nombre total d'événements en base
total_element	Nombre total d'éléments gérés

## Comment lire le check

### Taille de la base

Durant les 30 premiers jours d'activité du module, il est grandement recommandé de surveiller la taille de la base (avec la métrique : **total\_base\_size**), car la taille de la base ne fera que monter durant cette période.

Si la taille de la base se rapproche trop vite de la limite de votre disque, vous pouvez réduire le nombre de jours sauvegardés à l'aide de la clé : **day\_keept\_data** situé dans le fichier **/etc/shinken/modules/event\_manager\_writer.cfg** ou augmenter la capacité de votre disque.

Passé cette période, Shinken ne gardera que le x derniers jours définie par la clé **day\_keept\_data** afin de limité la taille de la base.



Une augmentation du nombre d'éléments supervisés fera grandir la taille de la base.

### Gestion du nombre d'événement écrits et du nombre de brok gérés

Le nombre d'événements doit être sensiblement inférieur au nombre de brok gérés, c'est pourquoi il faut surveiller les métriques **global\_brok\_handle\_in\_last\_min** et **global\_event\_write\_in\_last\_min**, car si c'est deux métriques sont proches cela signifie qu'à chaque vérifications, vos éléments changent d'état et donc que tout les éléments supervisés sont en flapping.

### Gestion des workers

#### Ajout d'un worker

Pour ajouter un worker, il suffit de modifier la clé **broker\_module\_nb\_workers** dans **/etc/shinken/modules/event\_manager\_writer.cfg** en augmentant ou diminuant le nombre de worker utilisé.

⚠ Chaque worker ajouté utilisera un CPU sur le serveur où se situe le démon Broker. Ajouter ou diminuer le nombre de worker permet de mieux répartir la charge de travailler pour les autres worker.

#### Gestion de la charge des workers

Conditions	Origine	Solution
Si les métrique <b>total_event_number</b> , <b>global_event_write_in_last_min</b> , <b>global_brok_handle_in_last_min</b> et <b>worker_[X]_load_in_last_min</b> croissent et que le temps de traitement des broks devient élevé	Il est probable que le nombre d'éléments supervisés ont augmentés	Il est alors conseillé d'augmenter le nombre de worker utilisés.
Si la métrique <b>total_event_number</b> est stable mais que la métrique <b>global_brok_handle_in_last_min</b> monte	Il est probable que vous avez changé le check intervalle sur vos checks	Surveiller la charge des workers et ajouter un si besoin.
Si la métrique <b>global_brok_handle_in_last_min</b> est stable mais que la métrique <b>global_event_write_in_last_min</b> monte	C'est que votre infrastructure passe une période d'instabilité (mise à jour sur les serveurs, changement de switch ...)	Surveiller la charge des workers et la taille de la base. Si le problème est temporaire la charge du worker va retrouver un niveau stable
Si la métrique <b>global_brok_handle_in_last_min</b> et <b>global_event_write_in_last_min</b> sont stable mais que la métrique <b>worker_[X]_load_in_last_min</b> monte	Il est possible que machine qui exécute Shinken a un problème (swap, stealing CPU ...)	Dans ce cas, lancer la commande <code>shinken-healthcheck</code> puis <code>top</code> afin de vérifier l'état de votre infrastructure shinken et des performance du serveur.